

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problems Mailbox.**

THIS PAGE BLANK (USPTO)

Process and program for a numerical control of a machine tool and a numerical control which works with means of this program

Patent number: EP0774701
Publication date: 1997-05-21
Inventor: ROTH PIERRE (CH); PAROZ CEDRIC (CH)
Applicant: TORNOS BECHLER SA (CH)
Classification:
 - **International:** G05B19/416
 - **European:** G05B19/416
Application number: EP19950810724 19951117
Priority number(s): EP19950810724 19951117

Also published as

EP077470

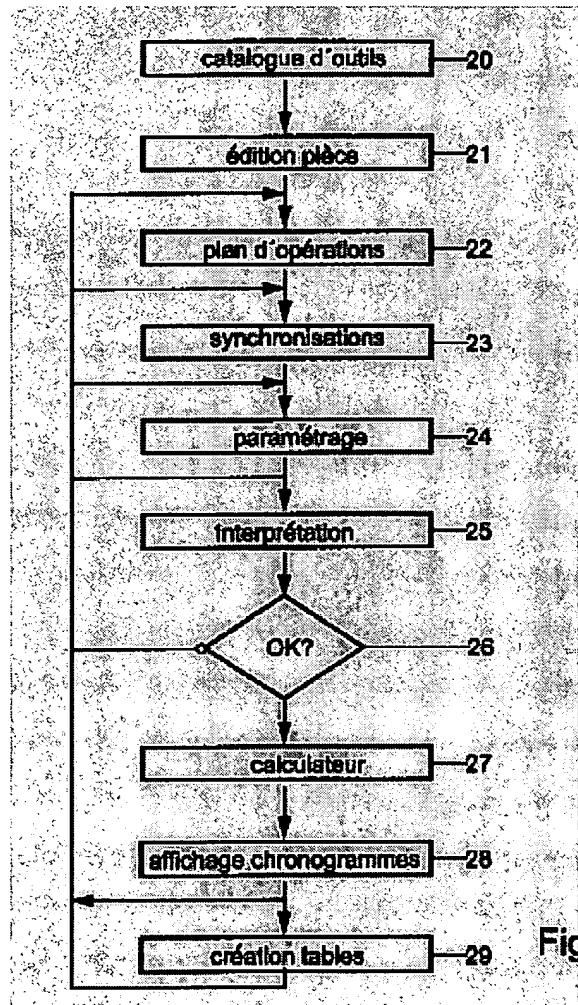
Cited documents:

EP047460
 US539220

Abstract of EP0774701

The procedure for generating step tables for digital control of a machine tool includes a preliminary step (20) of creating a catalogue of tools, allowing the operator to define the tools mounted on the machine tool. A further preliminary stage (21) allows the operator to provide parameters characteristic of the workpiece to be machined.

The next stage (22) allows definition of the channels and system axes, and introduction of a plan of the successive operations required. Synchronisation rules (23) are then imposed relating to the operations effected on different axes. The different operations are put into ISO or other computer language, and an operating program prepared. Step tables are then generated from this information.



Fic

THIS PAGE BLANK (USPTO)



(19)

Europäisches Patentamt

European Patent Office

Office européen des brevets



(11)

EP 0 774 701 A1

(12)

DEMANDE DE BREVET EUROPEEN

(43) Date de publication:
21.05.1997 Bulletin 1997/21(51) Int. Cl.⁶: G05B 19/416

(21) Numéro de dépôt: 95810724.5

(22) Date de dépôt: 17.11.1995

(84) Etats contractants désignés:
AT BE CH DE DK ES FR GB IT LI NL PT SE
Etats d'extension désignés:
LT LV SI(71) Demandeur: Tornos-Bechler SA
CH-2740 Moutier (CH)(72) Inventeurs:
• Roth, Pierre
2736 Sorvilier (CH)
• Paroz, Cedric
2732 Saicourt (CH)(74) Mandataire: BOVARD AG - Patentanwälte
Optingenstrasse 16
3000 Bern 25 (CH)

(54) Procédé et programme de commande numérique de machine-outil ainsi que commande numérique fonctionnant au moyen de ce programme

(57) Procédé et programme de génération de tables de pas pour commande numérique de machine-outil comprenant successivement:

une étape préliminaire (20) de création d'un catalogue d'outils permettant à l'opérateur de définir les outils (7, 7') montés sur la machine-outil.

une étape préliminaire (21) permettant à l'opérateur de paramétriser certaines caractéristiques de la pièce à usiner,

une étape (22) permettant à l'opérateur de définir tout d'abord plusieurs canaux ou systèmes d'axes, puis d'introduire un plan d'opérations constitué par une succession d'opérations consécutives dans chaque système d'axe, le plan d'opérations étant créé en plaçant avec la souris des symboles (46) sur une interface graphique (40),

une étape (23) permettant à l'opérateur d'imposer des règles de synchronisation entre opérations effectuées sur des systèmes d'axes différents (49, 50),

une étape (24) de paramétrage des différentes opérations au moyen du langage ISO ou d'assistants informatiques,

l'ensemble du plan d'opération et des règles de synchronisation constituant la gamme opératoire,

et une étape (25, 27, 29) de calcul des tables de pas à partir de la gamme opératoire.

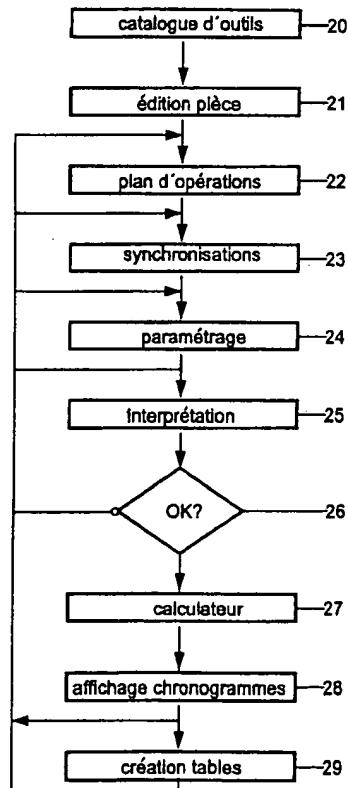


Fig. 6

Description

La présente invention concerne un procédé et un logiciel de génération d'ordres de déplacement pour une machine-outil. De manière générale, la présente invention concerne le problème de la commande numérique de machines-outils.

Art antérieur

La figure 1 illustre un exemple de machine-outil à commande numérique. Sur une plaque de base 1, un bâti 2 supporte une poupée 3 dont la broche est équipée d'une pince de serrage 4. Une barre à usiner 5 s'étend à l'avant de la poupée 3 et traverse un canon 6 de sorte que son extrémité se trouve en regard d'un premier outil de tournage 7, solidaire d'un premier coulisseau 8, et d'un deuxième outil de tournage 7' solidaire d'un deuxième coulisseau 8'. Un moteur de broche 9 fait tourner la pince 4. Un moteur de poupée 10 assure les déplacements vers l'avant et vers l'arrière de la poupée 3. Les commandes d'ouverture et de fermeture de la pince 4 ne sont pas représentées. Un moteur 11 déplace le premier coulisseau 8 sur une glissière 12, de manière à commander la position de l'outil 7 dans le sens radial par rapport à l'axe de rotation de la barre 5; de manière similaire, un moteur 11' déplace le deuxième 8' sur une autre glissière 12', de manière à commander la position du deuxième outil 7' dans le sens radial. Les pièces successives sont usinées en commandant les moteurs 10, 11 et 11' de manière à contrôler les déplacements des axes z, x et u. Dans certains cas, le moteur d'entraînement de la broche 9 peut également être commandé indépendamment.

Par la suite, on désignera les axes et les broches de la machine-outil par le terme général de ressources. Une ressource peut également être une combinaison mathématique virtuelle d'axes et de broches, comme on le verra plus loin.

Les différents moteurs sont contrôlés par le dispositif de commande numérique 13 à partir d'un programme généré sur un ordinateur externe 14 et transmis au dispositif de commande 13 par une ligne sérielle 15. Le programme de commande peut naturellement aussi être transmis autrement au dispositif 13, par exemple sur une disquette.

La figure 2 illustre le principe d'une commande numérique CNC traditionnelle dans le cas particulier d'une machine à trois axes.

Un interpolateur 61 reçoit des instructions 60 générées sur un ordinateur externe. L'interpolateur utilise ces consignes pour commander des déplacements des moteurs 65, 66, 67 correspondant à chaque ressource au travers des régulateurs 62, 63, 64. Des capteurs 68, 69, 70 mesurent en permanence la position ou la vitesse de chaque ressource; ces informations sont utilisées comme contre-réaction par les régulateurs correspondants. Les régulateurs 62, 63, 64 permettent d'assurer que les moteurs correspondants 65, 66, 67

suivent la consigne imposée par l'interpolateur 61 même en cas de perturbations.

Les instructions 60 sont idéalistes; par exemple, il peut y avoir des instructions de déplacements ou d'accélérations instantanés nécessitant une puissance infinie. L'interpolateur fournit par conséquent des consignes idéalistes aux régulateurs 62, 63, 64 et attend que les moteurs correspondants aient atteint la position demandée avant de commander le déplacement suivant. L'interpolateur utilise pour cela un feed-back fourni par les capteurs 68, 69, 70. Si des mouvements sont interpolés entre plusieurs ressources, l'interpolateur 61 attend que tous les moteurs concernés aient atteint la position demandée.

La fonction de l'interpolateur 61 est donc double: d'une part, il permet d'adapter les consignes de déplacement aux capacités physiques des moteurs 68, 69, 70, en fournissant de nouvelles consignes seulement lorsque les consignes précédentes ont été exécutées. D'autre part, il permet de réaliser des trajectoires multidimensionnelles impliquant des déplacements de plusieurs ressources, par exemple des déplacements rectilignes, circulaires ou paraboliques.

Les calculs complexes nécessaires à ces interpolations, qui sont effectuées en temps réel, exigent cependant une puissance de calcul importante qui renchérit considérablement les commandes numériques classiques. Par conséquent, les ressources susceptibles d'être utilisées ensemble dans une interpolation sont habituellement associées conceptuellement en plusieurs canaux. Une machine à 6 axes peut par exemple comporter 2 canaux formés chacun de trois axes.

Cette notion de canal limite considérablement les possibilités d'utilisation de la machine-outil, car une ressource appartenant à un canal ne peut pas être interpolée avec la ressource d'un autre canal. Il n'est par conséquent pas possible de tirer pleinement parti des fonctionnalités et de la rapidité de la machine-outil.

On peut dire que les machines-outils à commande numérique traditionnelles de la figure 2 fonctionnent en boucle fermée. En effet, l'interpolateur 61 calcule en temps réel les consignes appliquées aux régulateurs 62, 63, 64 en fonction des signaux fournis par les capteurs 68, 69, 70.

La plupart des interpolateurs 61 de l'art antérieur reconnaissent des instructions 60 dans une des nombreuses variantes du langage ISO. Ce langage comprend des instructions de haut niveau, impliquant de multiples déplacements, ainsi que des opérateurs de boucle ("for") ou de conditions ("if"). Il est possible de définir au moyen de ce langage une grande variété d'opérations à effectuer sur la pièce.

Le document de brevet EP474603 au nom de la déposante décrit une commande numérique fonctionnant selon un autre principe illustré par la figure 3. A chaque ressource de la machine correspond une table de pas 71, 72, 73 qui mémorise une suite d'ordres de déplacements pour cet axe. Ces déplacements sont imposés une fois pour toutes. Un compteur, non repré-

senté, provoque à intervalles déterminés la lecture dans chaque table de pas des incrément de déplacement de l'axe correspondant. Ces déplacements sont soit directement exécutés par des moteurs pas-à-pas 64', soit utilisés comme valeur de consigne par des régulateurs 62', 63' commandant des moteurs 65', 66' en utilisant des informations fournies par des capteurs 68', 60'. Il n'y a aucun feed-back entre les capteurs mesurant la pièce usinée et la commande numérique 13. On dit donc que le système fonctionne en boucle ouverte, la rétroaction depuis les capteurs 68', 69' ne s'appliquant qu'aux régulateurs associés 62', 63', mais n'influencant pas la séquence d'ordres de déplacements imposés.

Il y a une analogie entre le principe de cette commande et les machines traditionnelles, par exemple les décolleteuses, utilisant des cames. Dans ces machines, chaque outil est entraîné par une came. La trajectoire de chaque outil est donc provoquée par le déroulement du profil des cames; la position angulaire des cames détermine la position des outils. La synchronisation du mouvement des outils est assurée en montant toutes les cames sur un arbre à cames commun.

Pour chaque nouveau type de pièces à usiner, un nouveau jeu de cames doit être fabriqué. Le profil des cames doit ensuite être amélioré en les testant sur la machine et en modifiant leur profil tant que la pièce usinée ne correspond pas exactement à ce qui est désiré. La machine-outil reste donc généralement inemployée en production lorsqu'un nouveau programme de pièce est mis au point.

De manière correspondante, dans les machines à commande numérique décrites dans le brevet mentionné, la trajectoire de chaque outil est déterminée par le contenu des tables de pas correspondantes. Un nouveau jeu de tables de pas doit être créé au préalable pour chaque nouveau type de pièces à usiner. Comme tous les calculs d'interpolations sont calculés séparément et une seule fois sur un ordinateur 14 externe, la vitesse de calcul n'est pas critique; le calcul peut donc être complexe sans que la machine-outil soit ralentie.

L'usinage d'une pièce complexe pouvant impliquer des milliers de déplacements d'axes coordonnés, il n'est la plupart du temps pas possible de développer manuellement les tables de pas. Comme indiqué à la fin de la page 4 du brevet susmentionné, on crée donc préalablement ces tables au moyen d'un logiciel de calcul fonctionnant sur un ordinateur 14 externe à la machine.

La présente invention concerne notamment le problème de la génération des tables de pas 71, 72, 73 sur l'ordinateur externe 14. L'invention peut cependant aussi s'appliquer à la génération des instructions 60 pour un interpolateur dans une commande numérique traditionnelle selon la figure 2, ou à la génération d'ordres de déplacements dans une commande numérique quelconque.

On connaît dans l'art antérieur des systèmes permettant à l'opérateur d'introduire dans son logiciel de calcul le profil initial de la pièce ainsi que le profil désiré.

Un logiciel calcule alors automatiquement les ordres de déplacement des axes de manière à obtenir ce profil en un minimum d'opérations.

Ce type de logiciel est extrêmement compliqué à développer et offre généralement une souplesse limitée. Il peut parfois être difficile et lent d'introduire ou de définir exactement toutes les dimensions de la pièce désirée, en particulier si celle-ci a un profil compliqué. L'opérateur n'a en outre généralement aucune possibilité de modifier le plan d'opérations calculé si celui-ci ne lui donne pas entièrement satisfaction.

Pour toutes ces raisons, il est plus naturel d'introduire le profil de la pièce initiale, qui est de forme généralement simple, ainsi que les opérations d'usinage principales devant être effectuées sur cette pièce. L'opérateur contrôle lui-même les opérations d'usinage effectuées.

On pourrait utiliser à cet effet le langage ISO utilisé avec les commandes numériques traditionnelles selon la figure 1. Il est possible de définir au moyen de ce langage les opérations à effectuer sur la pièce, et de les convertir au moyen d'un compilateur en une succession d'instructions de déplacement des différents axes, regroupées dans les tables de pas.

Malheureusement, comme on l'a vu, les possibilités de description de ce langage sont insuffisantes pour commander avec la souplesse voulue des opérations d'usinage complexes. En particulier, le regroupement arbitraire des axes en canaux prédéfinis constitue une limitation qui empêche de tirer pleinement parti des possibilités de la machine. Il est en outre difficile avec ce langage d'introduire des synchronisations entre des déplacements d'axes appartenant à des canaux différents, et donc de définir avec la souplesse désirée des opérations d'usinage multitâches, c'est-à-dire faisant intervenir des déplacements simultanés de plusieurs systèmes d'outils.

Bien que de nombreuses extensions du langage ISO aient été proposées avec le développement de nouvelles commandes numériques, il est généralement difficile de synchroniser des opérations autrement qu'en ajoutant un nombre de phases d'attente suffisant après les opérations les plus rapides. Les machines-outils sont donc contraintes à fonctionner à une vitesse en dessous de leurs possibilités théoriques en raison de la faiblesse des possibilités de description du langage ISO. Ce problème est particulièrement crucial avec des machines-outils modernes comprenant plusieurs systèmes d'outils devant fonctionner en parallèle.

Le langage ISO est en outre devenu très compliqué à utiliser en raison de ces nouvelles instructions. L'opérateur qui généralement ne connaît pas la totalité du jeu d'instructions d'une nouvelle commande numérique renonce par conséquent bien souvent à des possibilités de programmation théoriquement disponibles.

L'utilisation du langage ISO ne permet donc que de générer des tables de pas insatisfaisantes, qu'il est nécessaire de tester directement sur la machine-outil pour vérifier Si les tables générées comportent des

phases d'attente inadmissibles. Dans ce cas, il est alors nécessaire de reprendre et d'améliorer les tables de pas, ce qui implique donc de nombreux essais successifs provoquant l'immobilisation de la machine.

5

Buts

Un but de la présente invention est donc de proposer un procédé et un logiciel de génération d'ordres de déplacement pour commande numérique de machine-outil évitant les inconvénients des dispositifs de l'art antérieur.

Un autre but de la présente invention est de proposer un procédé et un logiciel pouvant générer soit des ordres de déplacement, sous formes d'instructions, utilisables par des commandes numériques du type décrit en relation avec la figure 2, soit des ordres de déplacement implémentés sous formes de tables de pas utilisables par des commandes numériques du type décrit en relation avec la figure 3.

Un autre but est de proposer un procédé et un logiciel de génération d'ordres de déplacement adapté à n'importe quel type de machine-outil.

Un autre but de la présente invention est de proposer un procédé et un logiciel de génération d'ordres de déplacement pour une machine-outil facile à utiliser par l'opérateur. En particulier, un but de la présente invention est de proposer un procédé et un logiciel nécessitant le moins possible de connaissances spécialisées de la part de l'opérateur, et offrant un confort d'utilisation moderne.

Un autre but est de proposer un procédé et un logiciel de génération d'ordres de déplacement permettant aux machines-outils à commande numérique de fonctionner de manière beaucoup plus efficace qu'avec des ordres de déplacement générés à partir de logiciels utilisant des langages de description insuffisants tels que le langage ISO.

Un autre but est de proposer un procédé et un logiciel de génération d'ordres de déplacement permettant d'utiliser le plus efficacement possible les possibilités de la machine-outil à laquelle ces ordres sont destinés. En particulier, un but de la présente invention est d'éviter les limitations dues au regroupement arbitraire des ressources de la machine-outil en canaux prédéfinis.

Un autre but est de donner à l'opérateur un procédé et un logiciel de génération d'ordres de déplacement permettant de décrire et de générer des successions complexes d'opérations d'usinage synchronisées entre elles. En particulier, un but est de permettre à l'opérateur de définir lui-même les synchronisations de déplacements de ressources appartenant à différents systèmes d'axes sur la machine, de manière à utiliser pleinement les possibilités de travail multitâche de la machine-outil.

Un autre but de la présente invention est de proposer un procédé et un logiciel de génération d'ordres de déplacement pour une machine-outil qui laisse à l'opérateur la responsabilité de déterminer lui-même le plan

d'opérations.

Un autre but est de proposer un procédé et un logiciel de génération d'ordres de déplacement permettant de minimiser le nombre de tests nécessaires sur la machine.

10

Un autre but est de proposer un procédé et un logiciel de génération d'ordres de déplacements comprenant des moyens de simulation du processus d'usinage.

15

Ces buts sont atteints selon l'invention au moyen des éléments de la revendication 1.

20

Plus particulièrement, ces buts sont atteints par un procédé de génération d'ordres de déplacements pour une machine-outil dont les axes sont regroupés en une pluralité de systèmes d'axes, le procédé comprenant successivement:

une étape permettant à l'opérateur d'introduire dans chaque système d'axes un plan d'opérations constitué par une succession d'opérations consécutives, chaque opération étant choisie parmi un catalogue d'opérations,

une étape permettant à l'opérateur d'imposer des règles de synchronisation entre opérations effectuées dans des systèmes d'axes différents,

l'ensemble du plan d'opérations et des règles de synchronisation constituant la gamme opératoire,

et une étape de calcul des ordres de déplacement à partir de la gamme opératoire.

30

Dans une réalisation préférentielle, lesdits ordres de déplacement sont implémentés sous forme de tables de pas.

35

Dans une réalisation préférentielle, les ressources peuvent être librement regroupées en systèmes d'axes définissables par l'opérateur.

40

Dans une réalisation préférentielle, le procédé comprend en outre une étape de paramétrage des différentes opérations, permettant à l'opérateur de paramétrier les différentes opérations.

45

L'invention sera mieux comprise à l'aide de la description et des figures annexées parmi lesquelles:

50

- la figure 1 représente à titre d'exemple une vue de dessus d'une machine-outil de l'art antérieur pouvant fonctionner au moyen du procédé de la présente invention,
- la figure 2 illustre le principe de fonctionnement d'une commande numérique CNC traditionnelle,
- la figure 3 illustre le principe de fonctionnement d'une commande numérique utilisant des tables de pas,
- la figure 4 est un graphique illustrant un principe connu de l'élaboration d'une table de pas,

- la figure 5 montre schématiquement les données contenues dans une table de pas,
- la figure 6 est un organigramme du procédé et du logiciel selon la présente invention,
- la figure 7 est une vue d'écran d'une fenêtre d'édition d'outil dans un logiciel utilisé pour mettre en oeuvre le procédé de la présente invention,
- la figure 8 est une vue d'écran d'une fenêtre principale dans un logiciel utilisé pour mettre en oeuvre le procédé de la présente invention,
- la figure 9 illustre le principe de définition des synchronisations dans un logiciel utilisé pour mettre en oeuvre le procédé de la présente invention,
- la figure 10 est une vue d'écran d'une fenêtre de paramétrage des opérations dans un logiciel utilisé pour mettre en oeuvre le procédé de la présente invention, et
- la figure 11 est une vue d'écran d'une fenêtre de paramétrage des opérations dans un logiciel utilisé pour mettre en oeuvre le procédé de la présente invention.

Le graphique de la figure 4 illustre le principe de création manuel de tables de pas, correspondants aux axes z et x, et susceptibles d'être utilisées dans le dispositif de commande numérique 13 (figure 1). Le profil P d'une pièce que l'on désire usiner est représenté schématiquement dans le quadrant 1 en haut à droite du graphique. Dans ce graphique, les axes, indiqués par la référence t sont des axes temporels, quantifiés avec un pas de quantification fixe Δt valant par exemple 8 millisecondes. Les axes z et x sont gradués avec un pas correspondant à l'incrément des moteurs 9, respectivement 11.

Dans cet exemple, le profil P désiré est obtenu uniquement par déplacement des axes x et z, l'axe u restant immobile. L'opérateur indique tout d'abord le long du profil le temps nécessaire pour obtenir chaque section en fonction de la machine utilisée. A la fin de chaque intervalle de temps Δt , on reporte dans le quadrant 4 le déplacement Δz que doit avoir effectué la barre 5, et dans le quadrant 2 le déplacement Δx que doit avoir effectué l'outil 7 pour obtenir le profil souhaité. Sur la figure 6, on voit ainsi qu'à la fin du premier intervalle Δt le déplacement de la poupée doit être de 4 pas et celui de l'outil de 0 pas. La somme des intervalles de temps est la même pour l'axe z que pour l'axe y.

Les tables de pas correspondant à cet exemple sont illustrées sur la figure 5. Chaque colonne Δz , Δx , Δu représente le déplacement de l'axe correspondant après un intervalle Δt . Un compteur non représenté synchronise la lecture des tables de pas Δz , Δx , Δu à des intervalles Δt , les valeurs successives lues étant utili-

sées pour commander les moteurs 9, 11, 11' des axes correspondants. Ces tables permettent ainsi de programmer le dispositif de commande 13 de façon à usiner une pièce ayant le profil P sur la barre 5.

Dans une variante, les déplacements dans les tables de pas ne sont pas lus séquentiellement selon un ordre déterminé par un compteur, mais peuvent être accédés aléatoirement au moyen d'un registre d'adresse de lecture. L'adresse de lecture dans ce registre est incrémentée à chaque lecture de déplacement dans la table, mais peut aussi effectuer des sauts en fonction de l'état logique de la commande numérique.

Si la création manuelle ou même automatisée des tables de pas à partir du profil souhaité P est déjà fastidieuse dans un cas simple comme celui illustré, elle devient en pratique impossible pour commander des opérations complexes simultanées faisant chacune intervenir des mouvements interpolés et synchronisés entre plusieurs ressources. Il est donc nécessaire d'automatiser ce processus.

La figure 6 illustre à titre d'exemple un organigramme d'un logiciel de création des tables de pas selon l'invention.

Lors d'une première étape 20, l'opérateur définit au moyen du logiciel un catalogue d'outils. Il énumère dans ce catalogue tous les outils montés sur la machine-outil utilisée. Le logiciel affiche dans ce but une boîte de dialogue 30 telle que celle représentée sur la figure 7 pour éditer chaque outil. L'opérateur a la possibilité d'ajouter, de supprimer ou d'édition des outils dans son catalogue d'outil. La figure 7 illustre une boîte de dialogue permettant d'éditer un outil de coupe T15 nouvellement ajouté, c'est-à-dire de définir sa forme et sa position initiale. Les paramètres éditables dans la boîte de dialogue proposée dépendent de la nature de l'outil choisi, par exemple selon s'il s'agit d'un outil de coupe, de fraisage ou de perçage.

Lors d'une deuxième étape 21, l'opérateur définit au moyen d'une autre boîte de dialogue non représentée certains paramètres initiaux de la pièce à usiner. Ce paramétrage consiste à indiquer par exemple le diamètre et la longueur de la pièce initiale. De préférence, il est également possible d'entrer des données de type administratif permettant de retrouver la pièce, par exemple son numéro, le numéro d'identification des dessins correspondant à la pièce ou le nom de l'opérateur.

Lors d'une troisième étape 22, l'opérateur définit tout d'abord au moyen d'un menu ou d'une fenêtre de dialogue non représentée un ou plusieurs systèmes d'axes. En effet, selon une caractéristique importante de l'invention, les différentes ressources de la machine ne sont pas regroupées conceptuellement en canaux prédéterminés, mais peuvent être regroupées en systèmes d'axes librement définissables. Une ressource peut appartenir à plusieurs systèmes d'axes différents. Un système d'axes peut correspondre à une réalité physique, par exemple correspondre au plan de travail pour

un outil particulier, mais peut également représenter un espace virtuel. On peut ainsi définir si nécessaire des systèmes d'axes non orthogonaux, obtenus par combinaisons linéaires de ressources physiques. Il est aussi possible de définir des systèmes d'axes en mouvement, par exemple un système d'axes en rotation lié à la pièce à usiner. Cette possibilité permet de simplifier ultérieurement la description d'opérations sur cette pièce, par exemple de définir la vitesse de perçage dans un référentiel en rotation lié à la pièce usinée elle-même.

De préférence, le logiciel propose initialement des systèmes d'axes par défaut correspondant par exemple aux canaux habituels sur les machines à commande numériques CNC. Des options de menu non représentées permettent toutefois à l'opérateur de modifier ces systèmes d'axes par défaut ou d'ajouter de nouveaux systèmes d'axes. Des assistants informatiques sont de préférence utilisés pour définir des systèmes d'axes peu standards.

Ensuite, au cours de la même étape 22, l'opérateur définit le plan d'opérations pour chaque système d'axes. Le plan d'opérations pour chaque système d'axes est défini de préférence au moyen d'une fenêtre 40 illustrée sur la figure 8.

A chaque système d'axes défini correspond une ligne d'opérations 41 dans la fenêtre. Dans l'exemple illustré, il y a deux lignes d'opérations indiquées Peigne 1 et Peigne 2. Un symbole graphique 42 à gauche de chaque ligne d'opérations indique la nature du système d'axes. Dans l'exemple les systèmes d'axes correspondent aux axes de travail de deux chariots portant plusieurs outils de coupe. La liste des ressources 43 correspondant à chaque système d'axes est également indiquée à gauche. Le système d'axes correspondant au Peigne 1 utilise donc les ressources Z1, X1, Y1 et S1 tandis que le système d'axes correspondant au Peigne 2 utilise les ressources Z1, X2 et S1.

Une barre d'outil flottante 44 est affichée à droite de la fenêtre 40. En cliquant avec la souris sur une touche 45, correspondant à une opération d'usinage, dans la barre d'outil, l'opérateur sélectionne cette opération. Les opérations sélectionnables dans cet exemple sont le tournage externe (TE), le tournage interne (TI), le perçage/alésage (PA), le filetage (FI), le fraisage (FR), le formage (FO), le polygonage (PO) et les opérations auxiliaires (AO). En cliquant à nouveau dans une des lignes d'opérations 41, l'opérateur y insère un symbole graphique 46 représentant l'opération sélectionnée. Si le système d'axes de la ligne d'opérations sélectionnée ne peut pas effectuer l'opération choisie, un signal d'erreur sonore est émis par le logiciel et le symbole graphique 46 ne se dépose pas dans la ligne. En plaçant plusieurs symboles 46 à côté les uns des autres sur une ligne, l'opérateur choisit très facilement pour chaque système d'axes la succession d'opérations à effectuer et les place dans un ordre chronologique sans se soucier à ce moment de problèmes de durée ou de synchronisation. Il est possible de modifier très simplement l'ordre des opérations, en cliquant sur un symbole

graphique 46 et en relâchant la souris sur une nouvelle position.

Pour réaliser une série de pièces identiques sur une machine-outil, il est fréquent d'effectuer tout d'abord une séquence d'initialisation, puis de répéter dans une boucle une séquence d'usinage effectuée pour chaque pièce et après la dernière pièce d'effectuer encore une séquence de fin. L'opérateur peut définir très aisément le début et la fin d'une boucle d'opérations en plaçant avec la souris au-dessus des symboles graphiques 46 correspondant au début et à la fin de la boucle des fanions 47, 48 indiquant graphiquement le début et la fin de la boucle.

Lorsque l'opérateur a terminé la définition de son plan d'opérations dans les systèmes d'axes choisis, il impose au cours de l'étape 23 la manière dont les opérations sur les différents systèmes d'outils doivent être synchronisées. Les synchronisations peuvent être de type rigide ou extensible. Les synchronisations rigides sont des conditions strictes de synchronisme exact, par exemple:

"l'opération i doit se terminer en même temps que l'opération j". Au contraire, les synchronisations extensibles imposent des conditions moins strictes du type avant/après, par exemple:

"l'opération i doit se terminer avant le début de l'opération j". Ou bien:

"l'opération i doit se terminer 10 millisecondes après le début de l'opération j".

Il est possible de synchroniser soit le début, soit la fin des opérations, avec des événements tels que le début ou la fin d'une autre opération. Des options de menu appropriées permettent à l'opérateur de décaler le début et la fin d'une opération; par exemple, il peut décaler la fin d'une opération, sur laquelle se baseront les synchronisations des autres opérations, de 15 millisecondes par rapport à la fin effective de l'opération. L'opérateur peut aussi poser des bornes de synchronisation à l'intérieur d'une opération. Par exemple, il peut décomposer la durée d'une opération en 10 intervalles marqués par des bornes et synchroniser d'autres opérations par rapport à ces bornes. En outre, les opérations peuvent être synchronisées sur des signaux externes provenant par exemple de capteurs installés sur la machine ou sur des signaux d'horloge.

La figure 9 illustre le principe de placement des synchronisations rigides ou extensibles. L'opérateur sélectionne tout d'abord l'opération source devant être synchronisée. En cliquant dans la zone A à gauche du symbole graphique 46 correspondant à l'opération devant être synchronisée, il impose une condition sur le début de l'opération source; au contraire, en cliquant dans la zone B à droite du symbole graphique 46 correspondant à l'opération devant être synchronisée, il impose une condition sur la fin de l'opération source. Le curseur de la souris se transforme de préférence en un symbole différent selon la zone du symbole 46 sur

laquelle l'utilisateur a cliqué.

L'opérateur peut alors déplacer le curseur sur le symbole graphique 46 correspondant à l'opération destination. En cliquant dans la zone A1 en haut à gauche du symbole, il impose une synchronisation rigide de l'opération source sur le début de l'opération destination. En cliquant dans la zone B1 en haut à droite, il impose une synchronisation rigide sur la fin de l'opération destination. En cliquant dans la zone A2 en bas à gauche, il impose une synchronisation extensible sur le début de l'opération destination. En cliquant dans la zone B1 en bas à droite, il impose une synchronisation extensible sur la fin de l'opération destination. De préférence, un message d'aide s'affiche en bas de l'écran ou à proximité du curseur lorsque le curseur survole une des zones sensibles A1 à B2. La forme du curseur peut aussi être adaptée en conséquence.

D'autres manières de définir les synchronisations peuvent naturellement être prévues, par exemple en indiquant dans une boîte de dialogue l'opération source, l'opération destination ainsi que les conditions de synchronisation entre ces opérations.

Après avoir défini chaque synchronisation, celle-ci est indiquée visuellement, de préférence par une ligne forte 49 (figure 8) pour les synchronisations rigides et par une ligne 50 moins marquée ou d'une autre couleur pour les synchronisations extensibles. Dans l'exemple, un symbole triangulaire est en outre placé sur la ligne indiquant les synchronisations extensibles. Lorsque des synchronisations sont placées, les symboles graphiques 46 sur les différentes lignes 41 se déplacent de façon à indiquer au mieux la chronologie des opérations. En cliquant sur une ligne de synchronisation 49 ou 50, l'opérateur appelle une fenêtre de dialogue non représentée qui lui permet de définir des synchronisations plus complexes.

Le plan d'opérations défini dans les différents systèmes d'axes au cours de la troisième étape 22 ainsi que les règles de synchronisations définies au cours de la quatrième étape 23 définissent ensemble la gamme opératoire pour la pièce usinée. Au cours de la cinquième étape 24, l'opérateur peut paramétrier les opérations dont il a défini précédemment la chronologie. Cette étape permet d'indiquer exactement en quoi consistent les opérations choisies; par exemple, s'il a placé un symbole graphique 46 correspondant à une opération de fraisage, il peut au cours de l'étape 24 définir la profondeur de fraisage et l'endroit précis où le fraisage doit être effectué. Dans ce but, en cliquant deux fois rapidement sur le symbole graphique 46 de l'opération à paramétrier, l'opérateur voit s'afficher une fenêtre de dialogue 60 représentée sur la figure 10.

Dans le premier champ 61 de la fenêtre 60, l'opérateur peut indiquer le nom qu'il souhaite donner à l'opération. Ce nom peut dans une variante être affiché dans la partie inférieure des symboles graphiques 46. Par défaut, le logiciel propose un nom composé de deux nombres séparés par un ". ". Le premier nombre (5) identifie la ligne d'opérations tandis que le deuxième nom-

bre (2) identifie l'opération.

Dans une variante, un commentaire peut être introduit par l'utilisateur dans un masque approprié. Ce commentaire peut s'afficher par exemple dans la barre de statut sous la fenêtre lorsque le curseur de la souris survole le symbole graphique 46 correspondant à l'opération.

La deuxième ligne 62 de la fenêtre de dialogue 60 est une suite d'instructions non éditables en langage ISO. Ces instructions sont des instructions préliminaires dépendant de la nature de l'opération d'usinage.

Sous la deuxième ligne 62 se trouve une zone de texte éditable 63 dans laquelle l'opérateur peut introduire une suite d'instructions en langage ISO lui permettant de paramétrier et de définir l'opération à effectuer. De préférence, une suite d'instructions est proposée par défaut lorsque l'opérateur ouvre la fenêtre de dialogue. Des commentaires peuvent être ajoutés entre les instructions pour permettre même à l'utilisateur peu familier avec ce langage de paramétrier facilement toutes les opérations.

Sous la zone d'édition 63 est affichée une seconde suite 64 d'instructions non éditables en langage ISO. Ces instructions sont des instructions terminales dépendant de la nature de l'opération. Dans une variante, il n'y a pas d'instructions terminales.

Nous avons vu qu'une même ressource, par exemple un même axe, peut appartenir à plusieurs systèmes d'axes différents. Par exemple, sur la figure 8, l'axe Z1 appartient aussi bien au système d'axe du peigne 1 qu'au système d'axes du peigne 2. Suivant la gamme opératoire correspondant à ces deux systèmes d'axes, l'axe Z1 dans cet exemple pourrait être amené à devoir effectuer simultanément des déplacements dans des directions différentes; il est donc nécessaire de régler ces conflits potentiels. Pour cela, une zone 64 est affichée dans la fenêtre 60, qui permet de définir comment ces conflits éventuels doivent être réglés.

L'opérateur a la possibilité de supprimer une ressource commune, c'est-à-dire de la rendre non disponible pour le système d'axes considéré. Il peut aussi rendre une ressource virtuelle.

Dans ce dernier cas, si les deux opérations sur deux systèmes d'axes différents en conflit pour une ressource sont identiques et synchronisées de manière à démarrer simultanément, les déplacements de la ressource commune ne dépendent que de l'opération maîtresse, c'est-à-dire non virtuelle. L'opération virtuelle effectue une copie de l'opération maîtresse. Par exemple, dans le cas de la machine-outil décrite par la figure 1, des opérations utilisant l'outil 7 peuvent entrer en conflit pour le partage de l'axe z avec des opérations utilisant l'outil 7'. Si les opérations sont identiques et que la ressource z est déclarée virtuelle dans le système d'axes correspondant à l'outil 7', l'outil 7' effectuera simplement une copie miroir des déplacements de l'outil 7.

Si les deux opérations en conflit sont différentes, mais synchronisées de manière à démarrer et à se terminer simultanément, le logiciel calcule les déplace-

ments de la ressource commune en tenant uniquement compte de l'opération maîtresse. Les déplacements des autres ressources de l'opération virtuelle sont adaptés en fonction des déplacements ainsi imposés de la ressource commune. Dans l'exemple ci-dessus, si les opérations des outils 7 et 7' ne sont pas identiques, et que la ressource z est déclarée virtuelle pour le système d'axes correspondant à l'outil 7', le logiciel calculera la table de pas commandant l'axe z en fonction uniquement de l'opération sur l'outil z, et adaptera la table de pas commandant l'axe u portant l'outil 7' en fonction des mouvements prédéfinis de l'axe z.

Dans une variante préférentielle, des opérateurs connus dans les systèmes d'exploitation graphiques sous le nom "assistants" permettent à l'opérateur peu expert de paramétrier les opérations d'usinage sans même connaître le langage ISO. Ces assistants sont des modules de programme affichant une succession de boîtes de dialogue, dépendant de l'opération choisie, permettant à l'opérateur en répondant à des questions simples successives de paramétrier complètement l'opération. A chaque boîte de dialogue successive est associé une fenêtre d'aide expliquant exactement à l'opérateur les conséquences de ces choix. Ce type d'assistant est notamment connu en informatique dans les programmes d'installation de logiciels, et il n'est pas nécessaire de les décrire plus en détail. De préférence, les deux possibilités de paramétrage mentionnées sont combinées au cours de l'étape 24; l'utilisation de l'assistant génère automatiquement du code ISO qui est ensuite éditable.

Il est à noter que les étapes 22, 23 et 24 sur la figure 4 ne doivent pas être exécutées à la suite les unes des autres, mais le sont généralement selon un processus itératif, l'opérateur pouvant par exemple commencer à paramétrier ou synchroniser des opérations avant de les avoir toutes introduites.

Après avoir terminé de définir la gamme opératoire et avoir paramétré chaque opération, l'opérateur lance au cours de la sixième étape 25 l'interprétation du programme de pièce ainsi créé. Il peut effectuer cette opération en cliquant sur une option de menu appropriée dans la fenêtre de la figure 8. Durant cette étape d'interprétation, le logiciel détecte certaines fautes ou contradictions possibles dans la gamme opératoire, par exemple des conflits non résolus de ressources communes, des déplacements de ressource hors des limites de la machine ou des mouvements faisant intervenir des outils non installés ou des collisions d'outil. Le module d'interprétation détermine en outre au cours de cette étape 25 les séquences de déplacement des différentes ressources, définissant les positions de chaque ressource au début et au terme de chaque déplacement.

L'interpréteur calcule les séquences au moyen d'une table faisant correspondre à chaque instruction en langage ISO des portions de séquence prédéterminées.

Le logiciel vérifie au cours de l'étape 26 le résultat

fourni par l'interpréteur. En cas d'erreurs détectées, un message s'affiche sur l'écran et l'opérateur a ensuite la possibilité de modifier et de corriger la gamme opératoire. Si tout s'est passé correctement, l'opérateur peut lancer au cours de l'étape suivante 27 un calculateur qui calcule et affiche les trajectoires des outils ou des ressources au cours des séquences déterminées au cours de l'interprétation. Ces trajectoires sont directement utilisées pour la génération des tables de pas permettant de commander les moteurs des différents axes et broches.

Le fonctionnement du calculateur est connu dans l'art antérieur. Comme il relève des mathématiques et de la programmation pure, il n'est pas décrit ici.

La figure 11 illustre l'aspect de la fenêtre principale 40 affichée après l'étape 27 de calcul. La largeur des symboles graphiques 46 est modifiée de manière à correspondre à leur durée calculée, conformément à l'échelle des temps 51 affichée au-dessus des lignes de ressource. Un curseur 52 peut être déplacé au moyen de la souris pour mesurer de manière précise certains instants, le temps (3336 ms) s'affichant dans un rectangle tout en haut à gauche de la fenêtre. Dans la zone 52, un diagramme illustre la trajectoire de la ressource Z1 correspondante, c'est-à-dire la position de cet axe en fonction du temps. L'opérateur a naturellement la possibilité de choisir les ressources dont il souhaite contrôler la trajectoire. Il peut ainsi s'assurer que les déplacements des différents outils correspondent exactement à ce qui est désiré, et notamment que les outils occupent bien la même position au début et à la fin d'une boucle de répétition indiquée par les fanions 47 et 48.

Cette fenêtre permet ainsi à l'opérateur de contrôler très simplement, de manière visuelle, le déroulement des opérations successives d'usinage. En particulier, il est possible de se rendre compte aisément des possibilités d'accélérer le processus d'usinage en ajustant ou déplaçant les opérations les plus critiques. Si l'opérateur n'est pas satisfait par le résultat, il peut encore à cet instant revenir en arrière et modifier itérativement la gamme opératoire définie.

Selon les variantes, les tables de pas sont soit générées directement par le calculateur, soit obtenues d'après les données du calculateur au cours d'une étape 29 de création des tables. Ces tables sont ensuite transposées au moyen d'un support magnétique ou optique ou d'une ligne sérielle 15 sur la commande numérique 13 de la machine-outil (figure 1).

Le logiciel de l'invention permet donc de générer des tables de pas à partir du programme de pièce créé au cours des étapes 20 à 24. Il est cependant bien clair qu'il est possible de créer plusieurs programmes de pièces et d'enchaîner au cours de l'exécution la lecture des tables de pas générées par ces différents programmes. L'enchaînement des différentes tables de pas peut aussi dépendre de conditions externes. Un événement tel que le remplacement d'un outil, la rupture d'une mèche ou la fin d'une série de pièces par exemple peut

provoquer l'exécution d'une nouvelle table de pas.

Revendications

1. Procédé de génération d'ordres de déplacements pour une machine-outil dont les ressources (7, 7') sont regroupées en plusieurs systèmes d'axes, caractérisé en ce que le procédé comprend:

une étape (22) permettant à l'opérateur d'introduire dans chaque système d'axes un plan d'opérations constitué par une succession d'opérations consécutives, chaque opération étant choisie parmi un catalogue d'opérations,

une étape (23) permettant à l'opérateur d'imposer des règles de synchronisation entre opérations effectuées dans des systèmes d'axes différents,

l'ensemble du plan d'opérations et des règles de synchronisation constituant la gamme opératoire,

et une étape (25, 27, 29) de calcul des ordres de déplacement à partir de la gamme opératoire.

2. Procédé selon la revendication 1, caractérisé en ce que lesdits ordres de déplacement sont implémentés sous forme de tables de pas (71, 72, 73).

3. Procédé selon l'une des revendications précédentes, caractérisé en ce que les ressources peuvent être librement regroupées en systèmes d'axes définissables par l'opérateur.

4. Procédé selon l'une des revendications précédentes, caractérisé en ce que l'opérateur peut choisir parmi plusieurs types de règles de synchronisation entre opérations.

5. Procédé selon la revendication précédente, caractérisé en ce que l'une des règles de synchronisation possible consiste à définir qu'une opération doit commencer ou se terminer au moment précis où une autre opération sur un autre système d'axes commence ou se termine.

6. Procédé selon la revendication 4, caractérisé en ce que l'une des règles de synchronisation possible consiste à définir qu'une opération doit commencer au plus tôt ou se terminer au plus tard à l'instant où une autre opération sur un autre système d'axes commence ou se termine.

7. Procédé selon la revendication 4, caractérisé en ce que l'une des règles de synchronisation possible consiste à définir le début ou la fin d'une opération

par un délai à partir d'une autre opération.

8. Procédé selon la revendication 4, caractérisé en ce que l'une des règles de synchronisation possible consiste à définir le début ou la fin d'une opération par un événement externe fourni par un capteur sur la machine-outil.

9. Procédé selon l'une des revendications précédentes, caractérisé en ce qu'il comprend en outre une étape de paramétrage (24) des différentes opérations, permettant à l'opérateur de paramétrier les différentes opérations.

- 15 10. Procédé selon la revendication 9, caractérisé en ce que ladite étape de paramétrage (24) permet à l'opérateur de paramétrier chaque opération au moyen du langage ISO.

- 20 11. Procédé selon la revendication 9 ou 10, caractérisé en ce que ladite étape de paramétrage (24) permet à l'opérateur de paramétrier chaque opération au moyen d'assistants informatiques.

- 25 12. Procédé selon l'une des revendications précédentes, caractérisé en ce qu'il comprend en outre une étape préliminaire (21) permettant à l'opérateur de définir certaines caractéristiques de la pièce à usiner.

- 30 13. Procédé selon l'une des revendications précédentes, caractérisé en ce qu'il comprend en outre une étape préliminaire (20) de création d'un catalogue d'outils permettant à l'opérateur de définir les outils (7, 7') montés sur la machine-outil.

- 35 14. Procédé selon l'une des revendications précédentes, caractérisé en ce que ladite étape (25, 27, 29) de calcul des ordres de déplacement comprend une première étape (25) d'interprétation de la gamme opératoire au terme de laquelle l'opérateur est averti s'il a introduit des données erronées ou incompatibles entre elles ou avec la machine-outil utilisée.

- 40 45 15. Procédé selon la revendication précédente, caractérisé en ce qu'un test des collisions éventuelles entre outils est effectué au cours de ladite étape d'interprétation (25).

- 50 16. Procédé selon l'une des revendications précédentes, caractérisé en ce qu'une ressource peut être déclarée virtuelle dans un système d'axe lorsque cette ressource est simultanément utilisée dans un autre système d'axe.

- 55 17. Procédé selon l'une des revendications précédentes, caractérisé en ce que l'introduction par l'opérateur d'un plan d'opérations constitué par une

succession d'opérations consécutives se fait en déplaçant des symboles graphiques (46) sur une interface utilisateur graphique (40).

18. Procédé selon la revendication précédente, caractérisé en ce que les règles de synchronisation entre opérations sont introduites en alignant lesdits symboles graphiques (46), ou en les reliant par des éléments graphiques (49, 50). 5

19. Procédé selon l'une des revendications 17 ou 18, caractérisé en ce qu'au terme de ladite étape (25, 27, 29) de calcul des ordres de déplacement la largeur desdits symboles graphiques (46) est adaptée de manière à correspondre à la durée des opérations correspondantes. 10 15

20. Logiciel permettant de mettre en oeuvre le procédé de l'une des revendications précédentes.

21. Support magnétique ou optique sur lequel est enregistré un programme selon la revendication précédente. 20

22. Dispositif de commande numérique (13) pour machine-outil utilisant un logiciel selon la revendication 18. 25

30

35

40

45

50

55

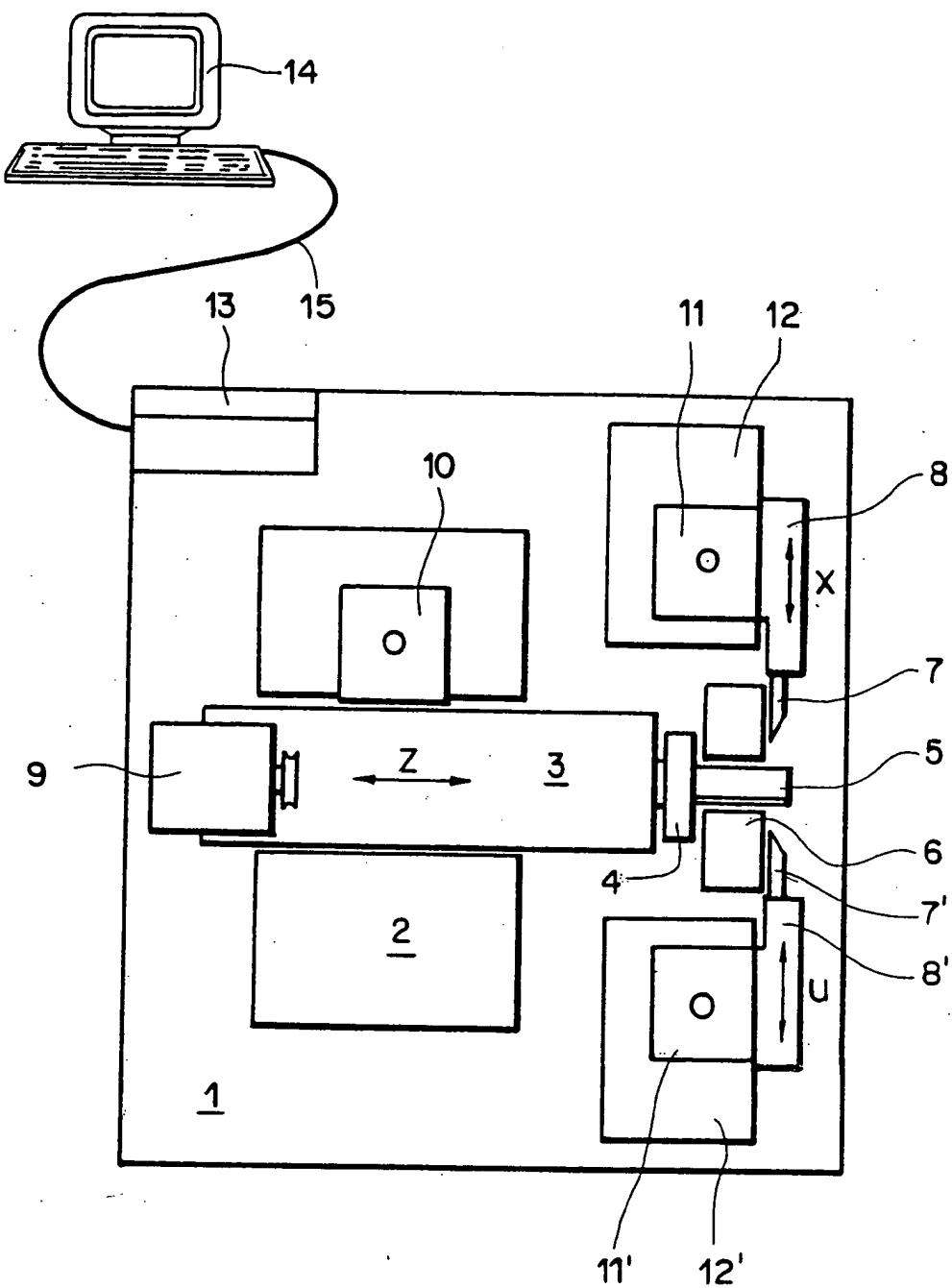
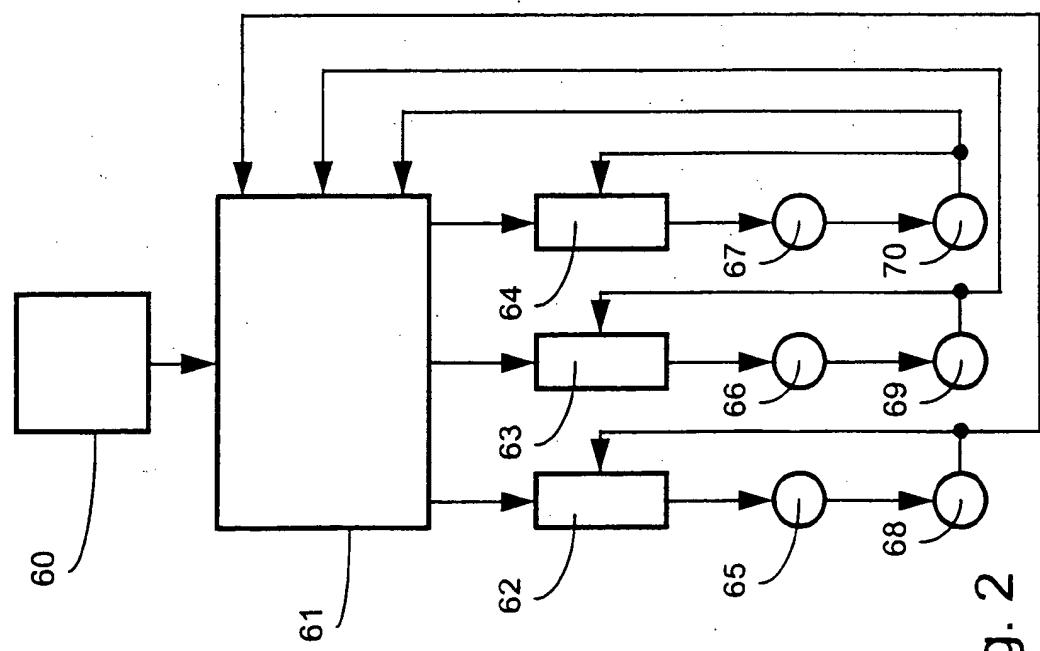
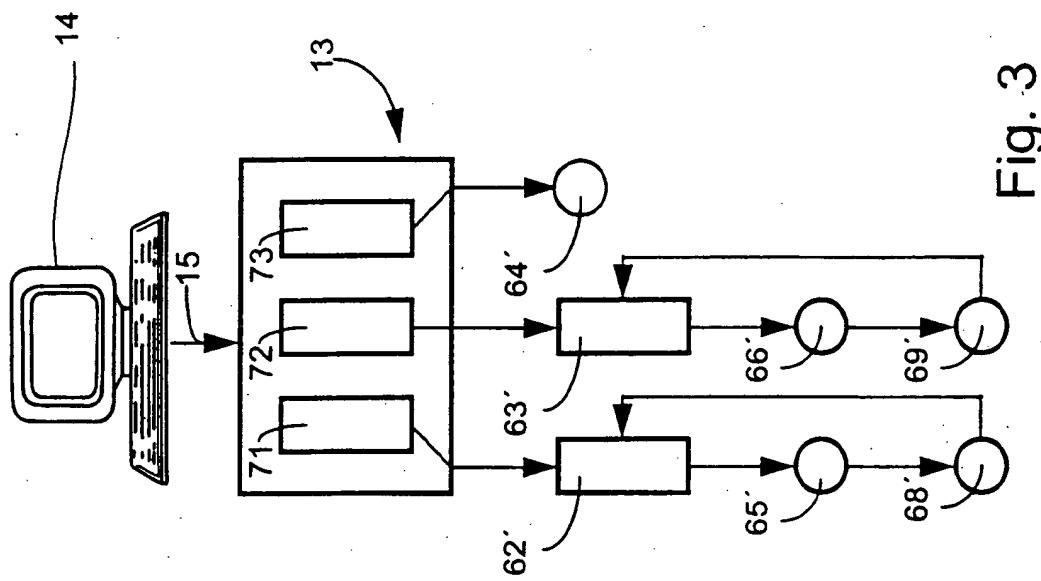


Fig. 1



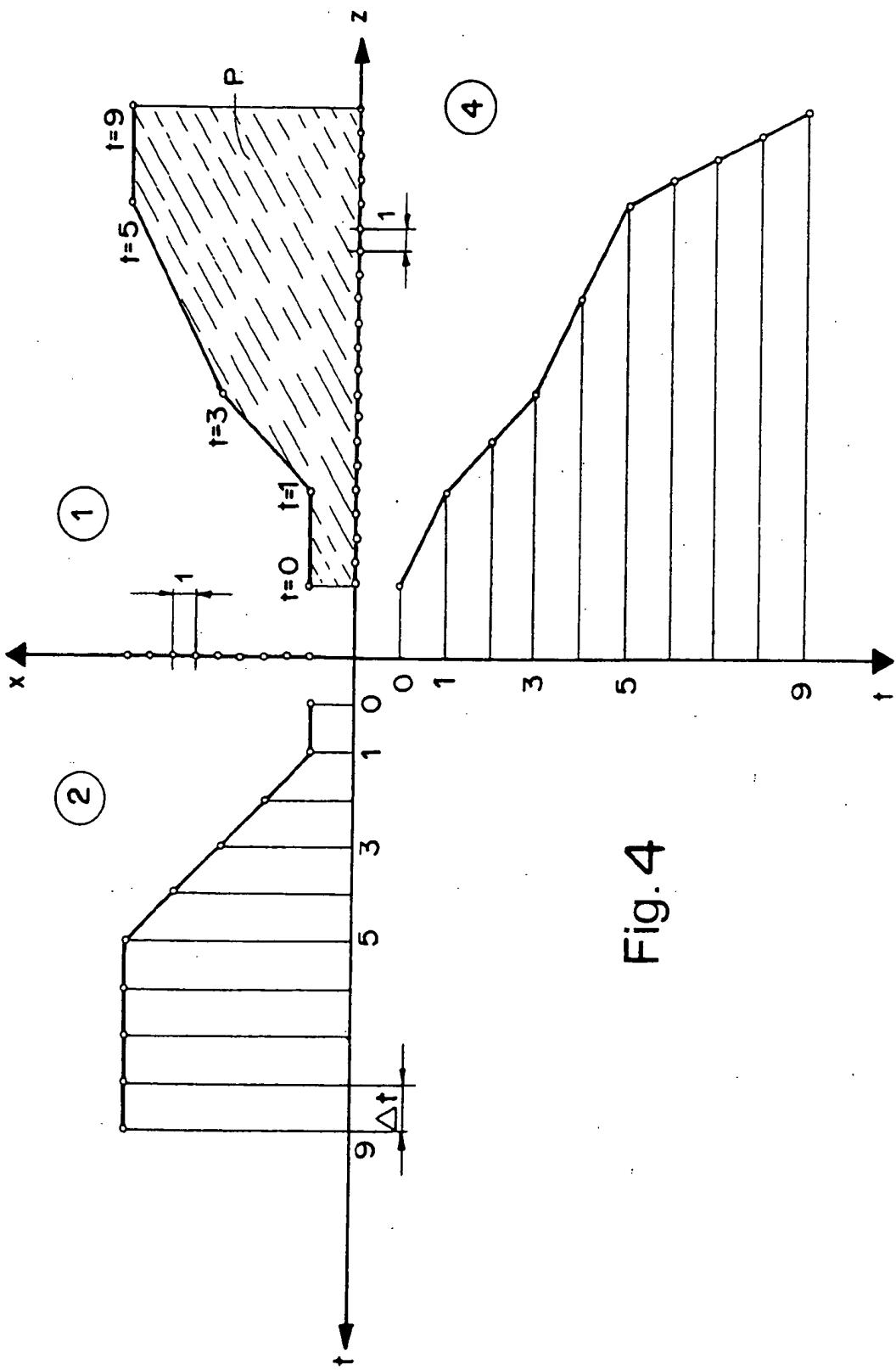


Fig. 4

| ΔZ | ΔX | ΔU |
|------------|------------|------------|
| 4 | 0 | 0 |
| 2 | 2 | 0 |
| 2 | 2 | 0 |
| 4 | 2 | 0 |
| 4 | 2 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |

Fig. 5

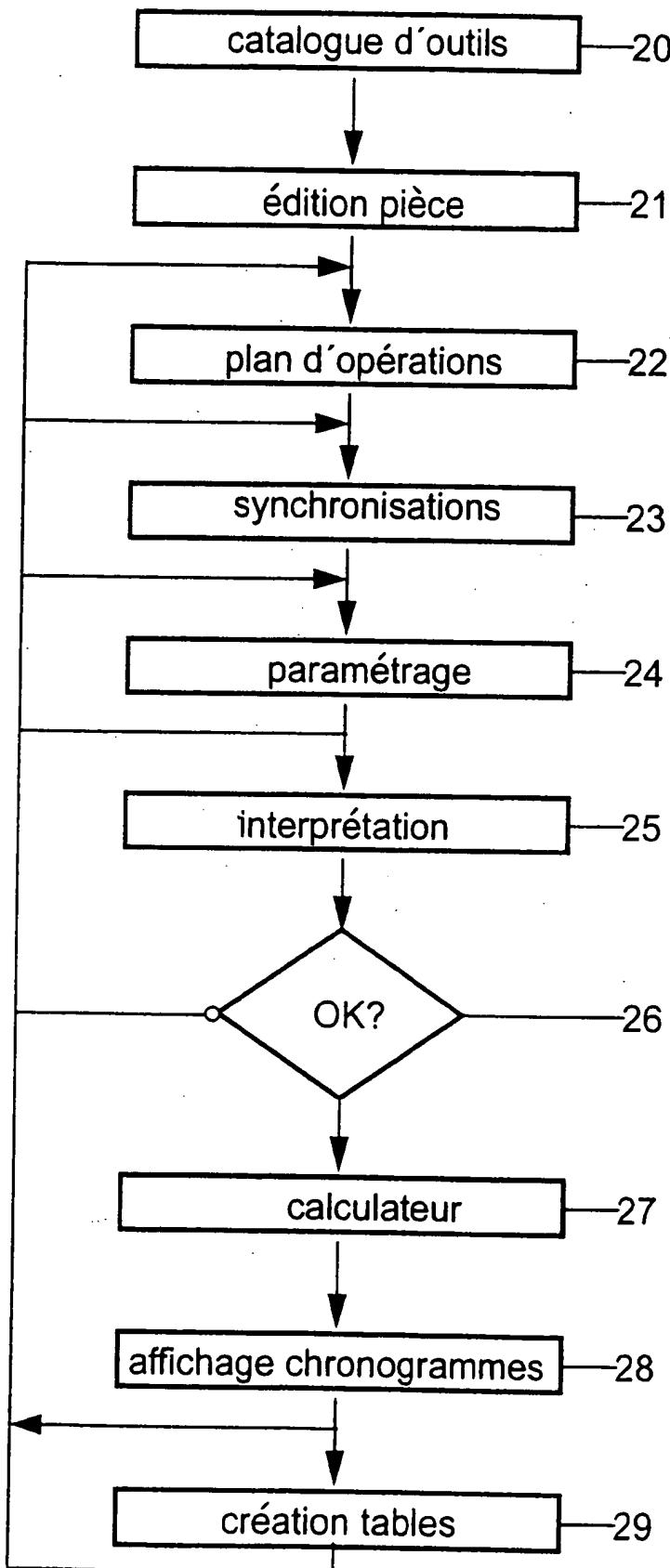


Fig. 6

Fig. 7

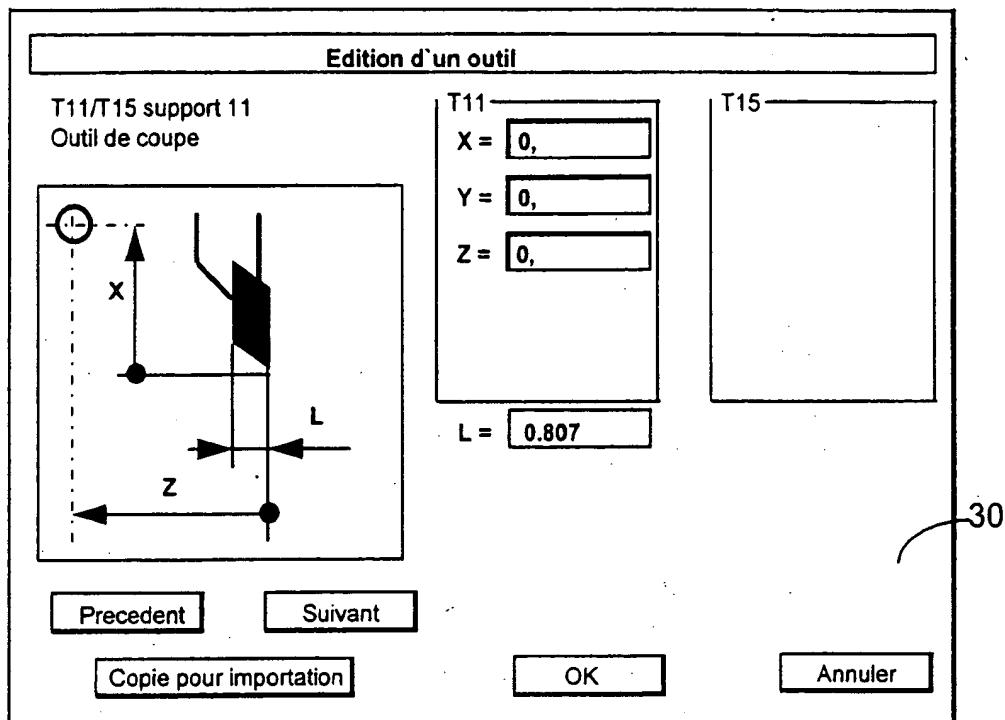
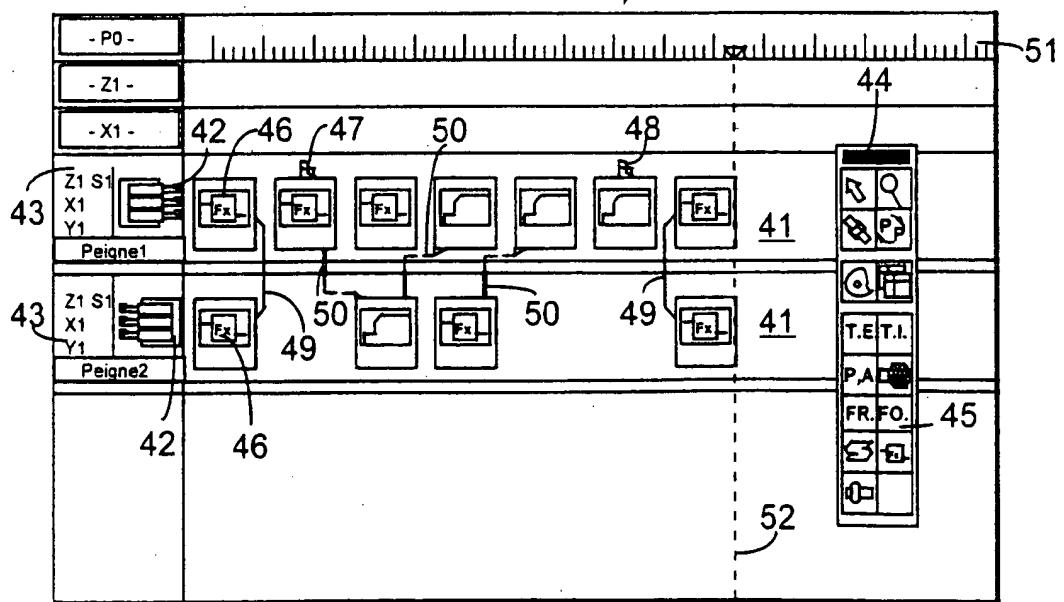


Fig. 8



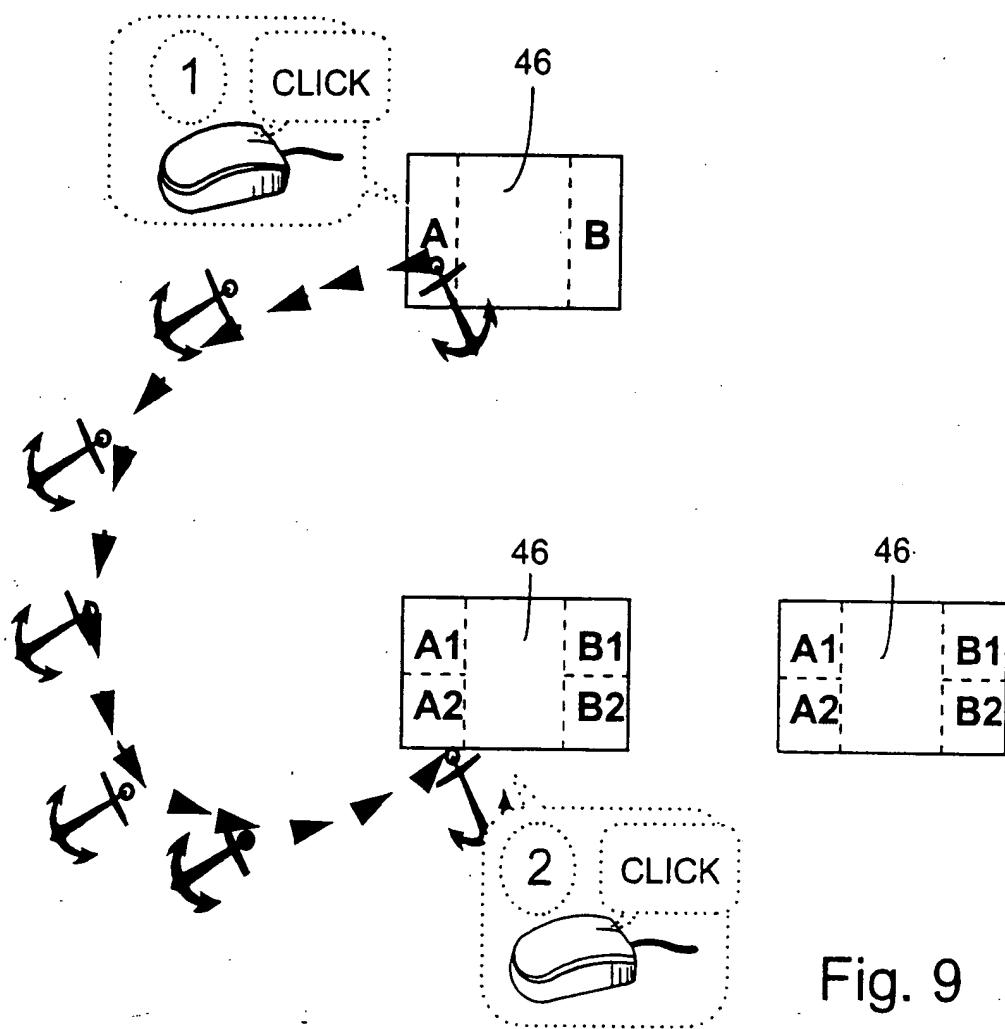


Fig. 9

Fig. 10

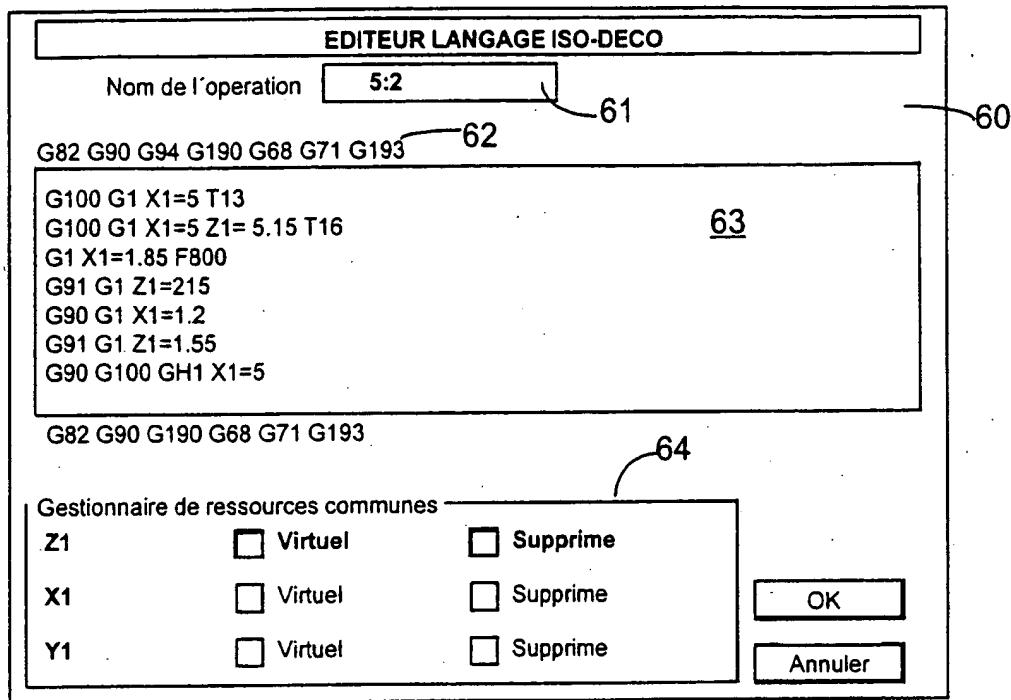
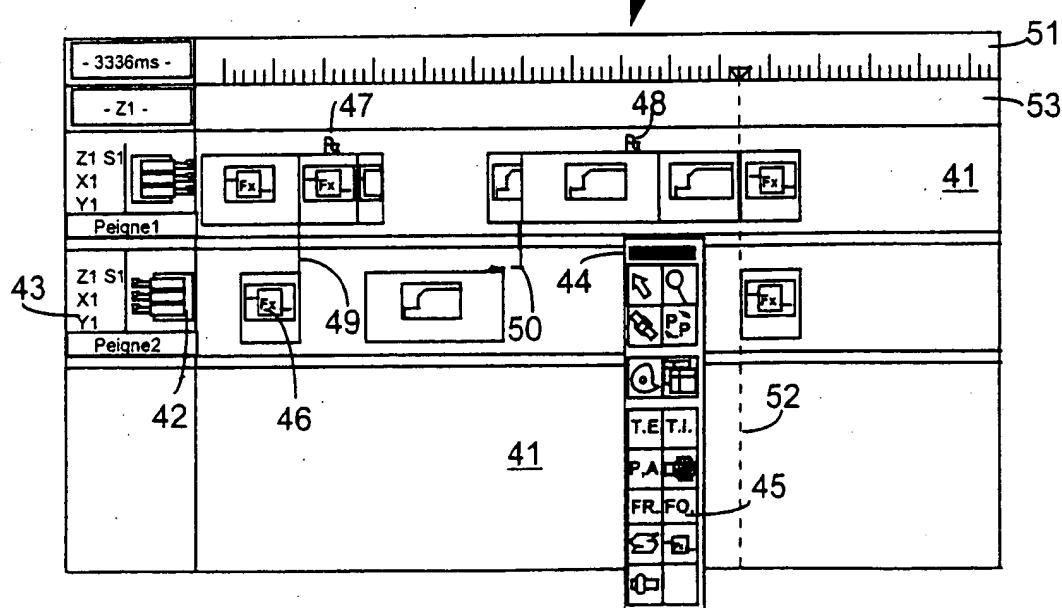


Fig. 11





Office européen
des brevets

RAPPORT DE RECHERCHE EUROPEENNE

Numéro de la demande
EP 95 81 0724

| DOCUMENTS CONSIDERES COMME PERTINENTS | | | | | | | | | |
|---|--|-------------------------|--|----------------------|-----------------------------------|-------------|---------|---------------|---------------|
| Catégorie | Citation du document avec indication, en cas de besoin, des parties pertinentes | Revendication concernée | CLASSEMENT DE LA DEMANDE (Int.Cl.6) | | | | | | |
| D,A | EP-A-0 474 603 (TORNOS SA FABRIQUE DE MACHINES MOUTIER) 11 Mars 1992 * colonne 3, ligne 5 - colonne 4, ligne 42; revendication 1; figures 1,2 * * abrégé * | 1-22 | G05B19/416 | | | | | | |
| A | US-A-5 392 207 (WILSON GREGORY S ET AL) 21 Février 1995 * colonne 5, ligne 42 - colonne 11, ligne 62; figures 1-4 * | 1-22 | | | | | | | |
| | | | DOMAINE TECHNIQUE RECHERCHÉ (Int.Cl.6) G05B | | | | | | |
| <p>Le présent rapport a été établi pour toutes les revendications</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 33%;">Lieu de la recherche</td> <td style="width: 33%;">Date d'achèvement de la recherche</td> <td style="width: 34%;">Examinateur</td> </tr> <tr> <td>LA HAYE</td> <td>18 Avril 1996</td> <td>Nettesheim, J</td> </tr> </table> <p>CATEGORIE DES DOCUMENTS CITES</p> <p>X : particulièrement pertinent à lui seul Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie A : arrête-plan technologique O : divulgation non-écrite P : document intercalaire</p> <p>T : théorie ou principe à la base de l'invention E : document de brevet antérieur, mais publié à la date de dépôt ou après cette date D : cité dans la demande L : cité pour d'autres raisons & : membre de la même famille, document correspondant</p> | | | | Lieu de la recherche | Date d'achèvement de la recherche | Examinateur | LA HAYE | 18 Avril 1996 | Nettesheim, J |
| Lieu de la recherche | Date d'achèvement de la recherche | Examinateur | | | | | | | |
| LA HAYE | 18 Avril 1996 | Nettesheim, J | | | | | | | |

THIS PAGE BLANK (USPTO)